Applicant : Ravi P. Singh et al.
Serial No. : 09/675,817
Filed : September 28, 2000
Page : 2 of 10

Attorney's Docket No.: 10559-284001 / P9291 - ADI
APD1803-1-US

Amendments to the Claims:

This listing of claims replaces all prior versions and listings of claims in the application:

Listing of Claims:

1.   (Currently amended) A method of aligning instructions in a processor comprising:

storing a plurality of instructions of different sizes in a plurality of buffer areas, each buffer area including a plurality of sub-buffers, each sub-buffer storing a unit instruction width, with an instruction of greater than a unit instruction width stored in more than one sub-buffer;

aligning a first instruction from said buffer areas;

decoding a size of the first instruction;

selecting at least one of said plurality of sub-buffers areas from which to output said first instruction on an output part;

during said outputting, first determin determining a beginning of a second instruction from selected ones of the plurality of sub-buffers based on the size of the first instruction, decoding the size of the second instruction, and second determining whether processing the second instruction will deplete said plurality of buffers buffer areas; and

based on said second determining whether processing the second instruction will deplete said plurality of buffer areas, instructing the plurality of buffers buffer areas to receive additional instructions.

2.   (Cancelled)

3.   (Currently amended) The method of Claim 1, further comprising comparing a most significant bit of a pointer to a first of the plurality of sub-buffer[[s]] to a most significant

Applicant : Ravi P. Singh et al.
Serial No. : 09/675,817
Filed : September 28, 2000
Page : 3 of 10

Attorney's Docket No.: 10559-284001 / P9291 - ADI
APD1803-1-US

bit of a pointer to a second ~~of the plurality of~~ sub-buffer[[s]] to determine whether processing one of the plurality of instructions will deplete ~~a~~ any of the buffer areas.

4.   (Currently amended) The method of Claim 1, further comprising storing a first instruction across a plurality of ~~storage elements~~ sub-buffers prior to processing the instructions.

5.   (Original) The method of Claim 1, further comprising adding the size of the first instruction to a current instruction position to determine the beginning of the second instruction.

6.   (Original) The method of Claim 1, further comprising aligning ahead a number of cycles equal to a cache latency.

7.   (Original) The method of Claim 1, further comprising aligning instructions in a digital signal processor.

8.   (Currently amended) The method of Claim 1, further comprising issuing a request to a memory to reload the plurality of ~~buffers~~ buffer areas.

9-17. (Cancelled)

18.   (Currently amended) The processor of Claim 27, wherein the transition detector compares a most significant bit of a pointer to a first ~~of the plurality of sub-buffers~~ subpart to a most significant bit of a pointer of a second ~~of the plurality of sub-buffers~~ subpart to determine whether processing one of the plurality of instructions will deplete ~~a~~ any of the buffer areas.

19.   (Previously presented) The processor of Claim 27, wherein the processor aligns ahead a number of cycles equal to a cache latency.

Applicant : Ravi P. Singh et al.
Serial No. : 09/675,817
Filed : September 28, 2000
Page : 4 of 10

Attorney's Docket No.: 10559-284001 / P9291 - ADI
APD1803-1-US

20. (Currently amended) The processor of Claim ~~21~~ 27, wherein the processor is a digital signal processor.

21. (Currently amended) An apparatus, including instructions residing on a machine-readable storage medium, for use in a machine system to align instructions in a processor, the instructions causing the machine to:

storing a plurality of instructions of different sizes in a plurality of buffer areas, each buffer area including a plurality of sub-buffers, each sub-buffer storing a unit instruction width, with an instruction of greater than a unit instruction width stored in more than one sub-buffer;

decode a size of a first instruction from said plurality of buffer areas;

select at least one of said plurality of sub-buffers ~~areas~~ from which to output said first instruction on an output part;

during said outputting, ~~first~~ determining a beginning of a second instruction from selected ones of the plurality of sub-buffers based on the size of the first instruction, decoding the size of the second instruction, and determining whether processing the second instruction will deplete ~~said~~ the plurality of ~~buffers~~ buffer areas; and

based on said ~~second~~ determining whether processing the second instruction will deplete the plurality of buffer areas, instruct the plurality of ~~buffers~~ buffer areas to receive additional instructions.

22. (Cancelled)

23. (Currently amended) The apparatus of Claim 21, wherein a most significant bit of a pointer to a first ~~of the plurality of~~ sub-buffer[[s]] is compared to a most significant bit of pointer to a second ~~of the plurality of~~ sub-buffer[[s]] to determine whether processing one of the plurality of instructions will deplete ~~a~~ any of the buffer areas.

Applicant : Ravi P. Singh et al.
Serial No. : 09/675,817
Filed : September 28, 2000
Page : 5 of 10

Attorney's Docket No.: 10559-284001 / P9291 - ADI
APD1803-1-US

24. (Currently amended) The apparatus of Claim 21, wherein a first instruction is stored across a plurality of ~~storage elements~~ <u>sub-buffers</u> prior to processing the instructions.

25. (Currently amended) A method of processing instructions within a processor, comprising:

storing instructions of different widths within a cache <u>having a plurality of buffer areas, each buffer area</u> having a plurality of subportions, each subportion in the cache storing a unit instruction width, where an instruction of unit width takes up a single subportion in the cache, and an instruction of more than said unit width takes up more than one ~~subportions~~ <u>subportion</u> within the cache;

multiplexing each of the <u>sub</u>portions of said cache to an output point, and selecting <u>at least</u> one of said cache <u>sub</u>portions as a current instruction;

during said selecting said current instruction, predicting which of said ~~instructions~~ <u>buffer areas</u> within said cache will be depleted of instruction data within a number of cycles approximately equal to a latency of the cache, and instructing loading of that number of ~~buffers~~ <u>buffer areas</u> with additional instruction information.

26. (Currently amended) A method as in claim 25, wherein said predicting comprises comparing a most significant bit of a pointer to a first subportion, to a most significant bit of a pointer to a second subportion, to determine if any of the ~~subportions~~ <u>buffer areas</u> will be depleted.

27. (Currently amended) A processor comprising:

a plurality of buffer areas, <u>each buffer area</u> adapted to store a plurality of instructions of different ~~width~~ <u>widths</u> in a plurality of subparts, each of said subparts storing a unit instruction width, and said instructions of greater than <u>a</u> unit instruction ~~widths~~ <u>width</u> being stored in multiple said subparts;

Applicant : Ravi P. Singh et al.
Serial No. : 09/675,817
Filed : September 28, 2000
Page : 6 of 10

Attorney's Docket No.: 10559-284001 / P9291 - ADI
APD1803-1-US

a multiplexer, connected to said plurality of subparts, and selecting and aligning <u>at least</u> one of said plurality of subparts from any of said subparts within said buffer areas as a current instruction; and

a predictor, operating to predict when at least one of the plurality of buffer areas will be empty, and to send a signal to instruct said at least one of the plurality of buffer areas to load another instruction data.